METODY ILOŚCIOWE W BADANIACH EKONOMICZNYCH
Tom XXVI/3, 2025, s. 131 – 140

MODEL GŁĘBOKIEGO UCZENIA LCA-NN NA BAZIE SYSTEMU POPRAWY PREDYKCJI ZA POMOCĄ DEKOMPOZYCJI WIELOWYMIAROWYCH

Ryszard Szupiluk https://orcid.org/0000-0002-4416-7131
Kolegium Analiz Ekonomicznych
Szkoła Główna Handlowa w Warszawie
e-mail: rszupi@sgh.waw.pl

Streszczenie: W niniejszym artykule przedstawimy system uczenia głębokiego łączącego sieci neuronowe MLP z wielowymiarowymi dekompozycjami. Stanowi on rozwinięcie systemu dla poprawy wyników predykcji w ujęciu wielomodelowym wykorzystującym metody ICA, PCA oraz NMF, co można potraktować także jako metodę agregacji modeli lub wielowymiarową filtrację komponentów zakłócających wyniki prognoz. Cały proces przedstawimy w nawiązaniu po problemu ślepej separacji wskazując na możliwe użyteczne inspiracje jak i istotne ograniczenia takich interpretacji.

Słowa kluczowe: predykcja, modele uczenia głębokiego, wielowymiarowe dekompozycje, sieci neuronowe

JEL classification: C02, C50

WPROWADZENIE

Modele uczenia głębokiego często powstają w efekcie połącznia różnych technik, metod lub algorytmów stosowanych wcześniej odrębnie, choć często w ramach jednej badania lub procesu [Mallat 2016]. Takie podejście zastosujemy także w niniejszym artykule do metody poprawy predykcji za pomocą dekompozycji wielowymiarowych [Szupiluk 2013; Szupiluk i in. 2014]. Metodę tą najpierw rozszerzymy o etap neuronowego douczania, a następnie, biorąc pod uwagę cały proces analityczny, uogólnimy do postaci uniwersalnego regresyjnego/ klasyfikacyjnego systemu głębokiego uczenia. W ramach prowadzonych rozważań przedstawimy różne możliwości rozwoju całej koncepcji oraz jej związki z problemem ślepej separacji.

https://doi.org/10.22630/MIBE.2025.26.3.12

Będący punktem wyjścia system poprawy predykcji (SPP) za pomocą dekompozycji wielowymiarowych polega na dekompozycji zbioru prognoz na ukryte komponenty wśród których identyfikowane są, wiązane z błędami prognozy, komponenty destrukcyjne. Ich eliminacja a następnie wykonanie dekompozycji odwrotnej, prowadzić może do poprawy wyników predykcji. Cały proces SPP można także intepretować lub porównywać z problemem ślepej separacji (ang. Blind Signal Separation-BSS) .

BSS jest to problem w którym odtwarzane są nieznane a priori sygnały źródłowe, zmieszane w także nieznanym systemie mieszającym [Cardoso 1998; Cichocki, Amari 2002]. Jednym z powodów analogii między SPP oraz BSS są stosowane w obydwu przypadkach dekompozycja takie jak ICA, PCA lub NMF [Cardoso 1998; Comon, Jutten 2010; Cichocki i in. 2009]. Choć można potraktować je jako czysto analityczne reprezentacje danych, to są one silnie związane z problemem ślepej separacji lub w takich kategoriach mogą być rozważane. Stąd dość naturalne nawiązania między omawianą metodą poprawy predykcji a problemem BSS.

Jednak głównym elementem niniejszej pracy jest generalizacja całej koncepcji na ogólny przypadek modelowania neuronowego z nauczycielem. W tym celu włączymy w ramy systemu także etap tworzenia samych modeli podstawowych (w SPP zakładaliśmy, że prognozy już posiadamy), zaś poszczególne etapy (warstwy) tego systemu będziemy starali się jak najbardziej uogólnić. W przypadku dekompozycji, będziemy się posługiwać pojęciem LCA (ang. Latent Component Analysis) mając na myśli dowolne metody identyfikujące ukryte komponenty o różnych charakterystykach. Stąd uogólniony system będzie określany jako LCA-NN.

WIELOWYMIAROWE ŁĄCZENIE MODELI

Podstawową całej koncepcji jest system poprawy predykcji. Jego konstrukcja jest następująca. Zakładamy istnienie zbioru m rezultatów predykcji $x_i(t)$ generowanych przez modele bazowe, gdzie $i = 1,...,m$ oznacza numer modelu, zaś t jest indeksem czasu lub numerem obserwacji. Wyniki predykcji zbierane są w jednej wielowymiarowej zmiennej $\mathbf{x}(t) = [x_1(t), x_2(t),...,x_m(t)]^T$, lub w zapisie macierzowym $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2,...,\mathbf{x}_m]^T$.

Zakładając, że wyniki predykcji są użyteczne i prezentują akceptowalną jakość, można stwierdzić, iż prognozowane wartości w pewnym stopniu odpowiadają wartościom rzeczywistym (obserwowanym), a w pewnym się różnią. Można więc powiedzieć, że dany rezultat predykcji jest mieszaniną pewnych ukrytych komponentów konstruktywnych $\hat{s}_j$, $j = 1,..., p$ oraz komponentów destrukcyjnych $\tilde{s}_l$, $l = 1,..., q$,. Komponenty konstruktywne, związane są z podobieństwem wartości przewidywanych i obserwowanych. Komponenty

destrukcyjne, odpowiadają za różnice lub błędy predykcji. Owe konstruktywne oraz destrukcyjne komponenty źródłowe mogą być mieszane w różny sposób. W przypadku liniowego modelu (systemu) mieszania mamy

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \tag{1}$$

gdzie macierz $\mathbf{A} = \left[a_{ij}\right] \in R^{m \times n}$ reprezentuje system mieszający zaś wektor

$$\mathbf{s}(t) = [s_1(t),...,s_n(t)]^T = [\hat{s}_1(t),...,\hat{s}_p(t),\tilde{s}_{p+1}(t)....,\tilde{s}_{p+q}(t)]^T, \tag{2}$$

gdzie, $n = p + q$ zawiera zestaw komponentów źródłowych.

Estymacja komponentów źródłowych jest kluczowym elementem całej koncepcji. Problem ten można przedstawić jako poszukiwanie transformacji odwrotnej do (1) tak, że

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t) \approx \mathbf{s}(t), \tag{3}$$

gdzie dla m=n macierz $\mathbf{W} = \mathbf{A}^{-1}$ pełni rolę systemu separującego odwrotnego do **A**.

Identyfikując system mieszający **A** oraz komponenty bazowe **s** oraz eliminując komponenty destrukcyjne (stawiając odpowiednio $\tilde{s}_l = 0$) otrzymuje się „poprawianą" wersją rezultatów predykcji x postaci

$$\hat{\mathbf{x}}(k) = \mathbf{A}[\hat{s}_1(t),...,\hat{s}_p(t),0_{p+1}(t)....,0_n(t)]^T, \tag{4}$$

gdzie $\hat{\mathbf{x}} = [\hat{x}_1,...,\hat{x}_m]^T$. Przy czym fakt poprawy jakości oraz jej stopień zależy od zawartości komponentów destrukcyjnych w poszczególnych prognozach. Ten sam efekt poprawy zostanie osiągnięty w wyniku zastąpienia w macierzy mieszającej $\mathbf{A} = \left[\mathbf{a}_1, \mathbf{a}_2,..., \mathbf{a}_n\right]$ odpowiedniej kolumny wektorami zer

$$\hat{\mathbf{A}} = \left[\mathbf{a}_1,..., \mathbf{a}_p, \mathbf{0}_{p+1}, \mathbf{0}_{p+2},..., \mathbf{0}_n\right], \tag{5}$$

co prowadzi do

$$\mathbf{x}(t) = \hat{\mathbf{A}}\mathbf{s}(t). \tag{6}$$

Kluczowym zagadnieniem w rozważanej procedurze, jest znalezienie takiej dekompozycji (transformacji), która prowadzi do dekompozycji danych x na komponenty konstruktywne i destrukcyjne. A następnie poprawne określenie, które komponenty są destrukcyjne. Pomijamy przy tym dyskusję o naturze tych komponentów w podziale na indywidualne czy wspólne dla modeli predykcyjnych. Indywidualne komponenty bazowe, związane ze specyfiką danego modelu będą dalej traktowane jako szczególny przypadek „komponentu wspólnego" o zerowym wpływie na wyniki innych modeli.

Naturalnymi propozycjami dekompozycji są metody związane z identyfikacją komponentów ukrytych, takie jak ICA, PCA, NMF, AMUSE itp. [Comon, Jutten 2010]. Przyjmiemy, że ogólnie możemy je zaliczyć do klasy metod LCA. Przy czysto mechanicznym użyciu tych metod, mamy do czynienia z zasadniczo analityczną reprezentacją danych. Uzyskanych tak komponentów, nie wiążmy z fizycznymi sygnałami lub interpretowalnymi modelami generowania danych. Ową fizyczną interpretację można jednak próbować wykonać, nawiązując do problemu ślepej separacji, gdyż tam także mamy do czynienia z estymacją nieznanych zmiennych (sygnałów, komponentów) zmieszanych w nieznanym systemie.

Przyjęcie interpretacji BSS oznacza założenie określonej charakterystyki sygnałów źródłowych oraz określonego sposobu ich mieszania. Od tych założeń zależy z kolei system separujący. Zakłada się, że przypadkowi liniowego mieszania powinien odpowiadać liniowy układ separujący, dynamicznemu mieszaniu dynamiczny separujący itp. [Cichocki 2013].

Jednak z praktycznego punktu widzenia, przyjmując określony model mieszania, należy mieć na uwadze, nie tylko zasadność założeń co do jego postaci, ale także istniejący stan wiedzy, na temat możliwości uzyskania skutecznego rozwiązania. Skuteczna estymacja komponentów źródłowych, na gruncie aktualnie znanych metod separacji, jest możliwa zasadniczo dla modelu liniowego [Hyvarinen i in. 2001; Mali, Mahajan 2024].

W większości algorytmów separacji, zakłada się także istotną relację między ilością sygnałów źródłowych a obserwowanych (zmieszanych). Przyjmuje się, że BSS jest możliwa gdy ilość sygnałów obserwowanych jest większa lub równa liczbie sygnałów źródłowych [Cardoso 1998]. W klasycznych algorytmach separacji jest to założenie kluczowe, odpowiadające za możliwość separacji [Araki i in. 2025]. Założenie to, o ile może to być spełnione w przypadkach technicznych lub fizycznych, to w przypadku danych z rynków finansowych sytuacja znacząco się komplikuje.

W przypadku danych giełdowych, trudno, a w zasadzie niemożliwe, jest ustalenie jakiegoś ograniczonego zbioru sygnałów źródłowych z którego w wyniku mieszania powstają wartości wszelkich instrumentów finansowych. Można więc przyjąć, że wszelkie separacje (dekompozycje, transformacje) mają tu charakter analityczny. Jednocześnie jednak, nie sposób wykluczyć sytuacji kiedy zastosowanie różnych dekompozycji do różnych zbiorów instrumentów (wartości aktualnych lub ich predykcji) prowadzi do „wyłowienia" podobnych lub identycznych komponentów (lub komponentu). Mogłoby to sugerować identyfikację pewnego fizycznego komponentu źródłowego związanego z „jakimiś" realnym procesem ekonomicznym. Jednak na bazie dotychczasowych badań taką sytuację należy uznać za stosunkowo mało prawdopodobną, zaś jej pojawienie się należy wiązać raczej z jakimś szczególnym przypadkiem, niż efektem systematycznych poszukiwań.

W efekcie można powiedzieć, że korzyści z analogii do problemu BSS wiążą się raczej z możliwościami użycia stosowanych tam metod niż z bezpośrednią

interpretacją estymacji komponentów źródłowych w kategoriach zagadnienia ślepej separacji.

AGREGACJA NEURONOWA

Opisana powyżej metoda może być rozważana i rozwijana w różnych aspektach. Przede wszystkim, wybór dekompozycji jest z reguły dość ograniczony i często motywowany ich popularnością w aspekcie teoretycznym oraz praktycznym. Należą one do standardowych metod analitycznych, zaś przesłanką do ich stosowania w prezentowanym systemie jest ich skuteczność w problemie BSS, który wydaje się najbliższą analogią dla naszego rozwiązania. Oznacza to, że w przypadku rozważanego systemu, podobnie jak w problemie separacji, stosowane dekompozycje mogą tylko pewnym stopniu odpowiadać „właściwym" formom estymacji komponentów źródłowych.

Kolejną kwestią jest identyfikacja komponentów destrukcyjnych. Metodą dającą jednoznaczne wyniki jest mechaniczna eliminacja każdego komponentu, dokonanie transformacji odwrotnej i ocena wyników prognozy. Przy czym należy mieć na uwadze, że możemy otrzymać komponenty o niejednoznacznym znaczeniu np. pojedyncze komponenty mogą mieć negatywny wpływ na wyniki predykcji ale ich kombinacje już pozytywny. Wykonalność takiego zadania jest ściśle związana z ilością badanych modeli predykcyjnych a w konsekwencji z ilością komponentów źródłowych. W efekcie, należy zbadań wpływ nie tylko eliminacji pojedynczych komponentów, ale wszystkich podzbiorów (2^n). W przypadku dużej liczby komponentów źródłowych możliwe są też aprioryczne oceny ich jakości [Szupiluk, Rubach 2017]

Jednak taka ostra (tak/nie) klasyfikacja komponentów źródłowych nie zawsze musi być optymalna. Określone komponenty także mieć konstruktywny wpływ na pewne modele a destrukcyjny na pozostałe. W takich przypadkach, prosta zero-jedynkowa interpretacja poszczególnych komponentów źródłowych może być niemożliwa. Także nie zawsze optymalna może być pełna eliminacja komponentu destrukcyjnego, czasem lepsze może być jego stłumienie (lecz nie zerowanie).

Ze względu na możliwą „nieostrą" klasyfikację danego komponentu źródłowego jako destrukcyjny lub konstruktywny, zastosujemy ich ważenie w pewnym ogólnym systemie mieszającym, którym zastąpimy transformację odwrotną. Będzie to system neuronowy z nauczycielem w postaci targetu z bazowych modeli predykcyjnych. Przyjmiemy, że będzie to sieć neuronowa typu MLP. Jednak podanie na wejście takiego systemu bezpośrednio komponentów źródłowych s oznacza w istocie czystą agregacje neuronową modeli bazowych transformowanych przez macierz wynikająca z dekompozycji separującej. Byłby to typowy preprocessing PCA lub ICA gdzie główną korzyścią jest dekorelacja (liniowa lub nieliniowa) danych wejściowych dla sieci neuronowej. Aby wykorzystać informacje pozyskane w wyniku identyfikacji komponentów destrukcyjnych przyjmiemy, że punktem startowym dla wag sieci neuronowej

(przynajmniej jednej z warstw) będzie macierz $\hat{\mathbf{A}}$ określona przez (5), czyli najlepsza macierz jaką można otrzymać poprzez pełną eliminację komponentów destrukcyjnych. Dla sieci typu wielowarstwowy perceptron (ang. Multilayer Perceptron -MLP) typu MLP(n,n,n)

$$\hat{\mathbf{x}} = \mathbf{g}_2(\mathbf{B}_2[\mathbf{g}_1(\mathbf{B}_1\mathbf{s} + \mathbf{b}_1)] + \mathbf{b}_2), \tag{7}$$

gdzie $\mathbf{g}_i(.)$ jest wektorem nieliniowości, $\mathbf{B}_i$ jest macierzą wag i $\mathbf{b}_i$ jest wektorem przesunięcia odpowiednio dla *i*-tej warstwy, *i*=1,2. W przypadku braku uczenia, sieć neuronowa (7) jest tożsama z (6), jeżeli ograniczymy się do jednej warstwy $\mathbf{B}_1 = \hat{\mathbf{A}}$, $\mathbf{b}_i = \mathbf{0}$ oraz liniowych funkcji aktywacji. Jest jednak wysoce prawdopodobne, że jeżeli uczenie sieci neuronowej zaczniemy od punktu do którego doprowadza nas etap dekompozycji (wartości początkowe pierwszej warstwy ukrytej $\mathbf{B}_1(0) = \hat{\mathbf{A}}$) to osiągnięte wyniki będą lepsze niż z samej rekompozycji (dekompozycji odwrotnej). Możemy ten sposób rozumowania rozszerzyć na kolejne warstwy oraz przyjąć, że wagi we wszystkich warstwach startują od wartości $\hat{\mathbf{A}}$, czyli mamy

$$\hat{\mathbf{x}} = \mathbf{g}_2(\mathbf{B}_2[\mathbf{g}_1(\mathbf{B}_1\mathbf{s})]), \quad \text{gdzie } \mathbf{B}_1(0) = \mathbf{B}_2(0) = \hat{\mathbf{A}}. \tag{8}$$

Dalsze uogólnienie może polegać na dowolnej architekturze MLP wykorzystującej macierz $\hat{\mathbf{A}}$ oraz dane wejściowe **s**

$$\hat{\mathbf{x}} = \mathrm{MLP}(\mathbf{s}, \hat{\mathbf{A}}), \tag{9}$$

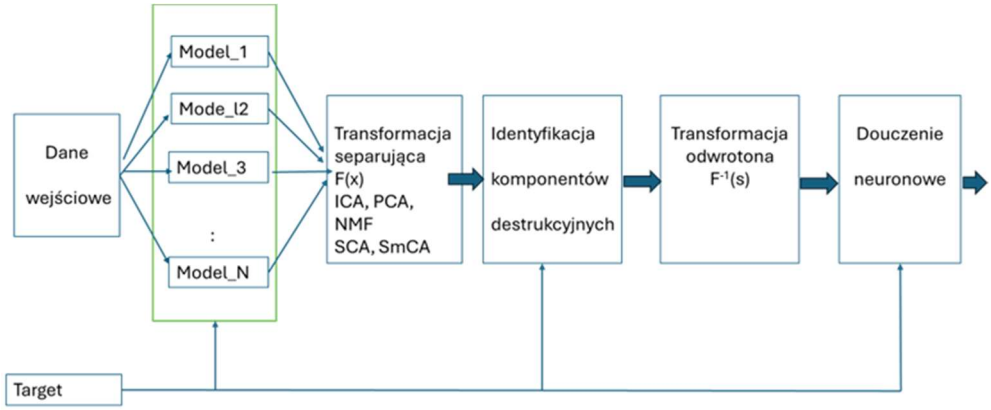
gdzie formuła (9) oznacza wykorzystanie dowolnej architektury MLP z sygnałami wejściowymi **s** (komponenty źródłowe) oraz dowolnymi warstwami wag startującymi od wartości macierzy $\hat{\mathbf{A}}$. Dalsze modyfikacje mogą polegać na wykorzystaniu innych typów sieci neuronowych np. RBF, CNN. LSTM.

MODEL UCZENIA GŁĘBOKIEGO LCA-NN

W dotychczasowych rozważaniach bazowaliśmy na założeniu, że już posiadamy wyniki z modeli predykcyjnych. Obecnie także ten etap, tworzenia modeli predykcyjnych oraz generowania przez nie wyników, włączymy do naszych rozważań. Analizując cały proces można zauważyć, że otrzymamy pełny system modelowania danych (regresja, klasyfikacja), który może spełniać różne role i nie musi się ograniczać do poprawy predykcji. Może on stanowić uniwersalny model neuronowy z nauczycielem, w którym występują etapy (lub inaczej warstwy): tworzenia podstawowych modeli regresyjnych lub klasyfikacyjnych (predykcje), następnie etap dekompozycji odpowiedzialny za oczyszczenie uzyskanych wyników z komponentów zakłócających i kolejny etap neuronowy mający za zadanie

douczanie całego systemu. Ze względu na możliwość użycia różnych dekompozycji prowadzących do różnego typu ukrytych komponentów źródłowych, ten aspekt funkcjonowania systemu określimy mianem LCA (ang, Latent Component Anlaysis), zaś cały system, prezentowany na rysunku 1, zdefiniujemy jako LCA-NN.

Rysunek 1. Sieć neuronowa LCA-NN



Źródło: opracowanie własne

Cały proces ma następującą postać:

Warstwa 1. Tworzymy modele predykcyjne w oparciu o sieci neuronowe MLP. Zestaw wyników z modeli neuronowych zbieramy w jednej zmiennej wielowymiarowej **x**.

Warstwa 2. Wykonujemy dekompozycję LCA (ICA, PCA, SMA, NMF, AMUSE itp.). Zmienną **x** poddajemy transformacji LCA otrzymując komponenty źródłowe **s**

Warstwa 3. Identyfikujemy komponenty destrukcyjne.

Warstwa 4. Dokonujemy uogólnionej transformacji odwrotnej. Zastępujemy, wartości komponentu destrukcyjnego zerami lub niwelujemy jego wpływ odpowiednimi parametrami w transformacja odwrotnej.

Warstwa 5. Wykonujemy douczanie neuronowe siecią MLP. Wartości początkowe warstw sieci MLP stanowią wartości z macierzy transformacji odwrotnej z zerowymi wartościami dla zidentyfikowanych komponentów destrukcyjnych.

Wyjście. Otrzymujemy sygnały wyjściowe.

Ze względu na wykorzystanie metod LCA, cała metoda jest domyślnie wielomodelowa, jednak nie oznacza to konieczności tworzenia całkowicie odrębnych modeli. Możliwe jest generowanie wielu rozwiązań tym samym modelem (to jest o tej samej architekturze, a także parametrach) na różnych podzbiorach

uczących. W efekcie całe rozwiązanie można także rozpatrywać w kategoriach metody agregacji modeli (ang. ensemble methods).

BADANIE PRAKTYCZNE

Niniejszy artykuł ma główne charakter ideowy oraz metodologiczny. Jednak aby pokazać zasadność i skuteczność podstawowej koncepcji przedstawimy krótko eksperyment przewidywania zużycia energii elektrycznej sformułowany jako predykcja godzinnego obciążenia dokonywana z wyprzedzeniem 24 godzin. Dane wejściowe dla predykcyjnych modeli neuronowych obejmują informacje o godzinnym zużyciu energii w ciągu ostatnich 24 godzin, jak również dane kalendarzowe typu: dzień, miesiąc, dzień tygodnia, indykator dnia. Dla tak postawionego problemu, opracowano pięć modeli neuronowych typu MLP o 20, 26, 28, 30, 35 neuronach w warstwie ukrytej, oznaczonych odpowiedni jako M1:MLP20, M2:MLP26, M3:MLP28, M4:MLP30, M5:MLP35.

W ramach badań wykorzystano dekompozycje PCA oraz ICA (algorytm Jade). Badanie przeprowadzono z wykorzystaniem samych dekompozycji jak i douczania neuronowego. Uzyskane wyniki prezentuje Tabela 1.

Tabela 1. Prognozowanie zużycia energii-wyniki modeli

MAPE 10^{-3}	M1	M2	M3	M4	M5
Modele podstawowe	23,73	23,44	23,71	24,01	24,13
PCA	22,87	22,45	22,79	22,57	22,51
ICA	23,69	22,41	22,57	23,34	22,98
PCA&NN	22,53	22,20	22,49	22,48	22,21
ICA&NN	22,17	22,38	22,51	22,71	21,16

Źródło: opracowanie własne

Jako kryterium jakości przyjęto błąd $MAPE = \frac{1}{N} \cdot \sum_{i=1}^{N} \frac{|\varepsilon_i|}{s_i}$, gdzie s_i oznacza wartość rzeczywistą (target) zaś ε_ibłąd prognozy. W większości modeli uzyskano poprawę już przy wykorzystaniu samych dekompozycji na poziomie od 3,6 do 6,7 procent. W przypadku agregacji neuronowej poprawy jakości predykcji wynoszą od 5,0 do 12,03 procent.

PODSUMOWANIE

Prezentowane w artykule rozważania odnoszą się do koncepcji, która bazuje na systemie poprawy predykcji, dla której zaproponowaliśmy szersze rozwinięcia neuronowe. Takie podejście otwiera także drogę do bardziej ogólnego punktu widzenia i wyjścia poza schemat systemu poprawy predykcji, np. potraktowanie

całego procesu jako uniwersalnego sytemu neuronowego z nauczycielem, który może być stosowany do dowolnego zagadnienia klasyfikacji lub regresji. Złożoność całego systemu uzasadnia określenie go jako system głębokiego uczenia.

Poszczególne etapy związane z dekompozycjami wielowymiarowymi oraz identyfikacją komponentów destrukcyjnych mogą być rozważane w kategoriach problemu ślepej separacji dając możliwość lepszego dopasowania stosowanych dekompozycji do badanego problemu oraz głębszej jego interpretacji. Wiązanie jednak działania dekompozycji z BSS nie jest jednak konieczne ani czasem możliwe, w takiej sytuacji traktujemy te przekształcenia LCA w kategoriach reprezentacji analitycznych.

Zarówno w przypadku systemu poprawy predykcji jak i sytemu LCA-NN, ze względu na identyfikację ukrytych komponentów możemy wiązać ten etap (warstwę) z problemem ślepej separacji, wykorzystując rozwiązania z tego obszaru, Należy jednak mieć na uwadze, że zbyt bliskie wiązanie podstawowego procesu poprawy predykcji oraz LCA-NN z BSS niesie z sobą także liczne trudności interpretacyjne i metodyczne, które trudno przekonująco rozstrzygać lub wprost akceptować. Także wprowadzeni etapu agregacji neuronowej, oddala cały proces od bezpośredniej interpretacji w kategoriach BSS.

BIBLIOGRAFIA

Araki S., Ito N., Haeb-Umbach R. et al. (2025) 30+ Years of Source Separation Research: Achievements and Future Challenges. Technical Report No. TR2025-036. Mitsubishi Electric Research Laboratories.

Cardoso J. (1998) Blind Signal Separation: Statistical Principles. Proceedings. of the IEEE, 86(10), 2009–2025.

Cichocki A., Amari S. (2002) Adaptive Blind Signal and Image Processing, John Wiley, Chichester.

Cichocki A., Zdunek R., Phan A.-H., Amari S. (2009) Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis. John Wiley. https://doi.org/10.1002/9780470747278

Cichocki A. (2013) Tensor Decompositions: A New Concept in Brain Data Analysis? 02 May 2013 - arXiv: Numerical Analysis

Comon P., Jutten Ch. (2010) Handbook of Blind Source Separation: Independent Component Analysis and Applications, Academic Press.

Hyvarinen A., Karhunen J., and Oja E. (2001) Independent Component Analysis. Wiley John, New York.

Mali S.G., Mahajan S.P. (2024) Blind Sound Source Separation by Combining the Convolutional Neural Network and Degree Separator. Traitement du Signal, 41(3), 1429-1439.

Mallat S. (2016) Understanding Deep Convolutional Networks. Philosophical Transactions of the Royal Society A 374, 2065,

Szupiluk R., Wojewnik P., Ząbkowski T. (2014) Smooth Component Analysis and MSE Decomposition for Ensemble Methods in Multi-Agent Environment. International Journal of Innovative Computing Information and Control, 10(4), 1435-1445.

Szupiluk R., Rubach P. (2017) Identyfikacja komponentów destrukcyjnych w modelach predykcyjnych w podejściu wielomodelowym. Metody Ilościowe w Badaniach Ekonomicznych, 18(4), 679-688.

Szupiluk R. (2013) Dekompozycje wielowymiarowe w agregacji predykcyjnych modeli Data Mining. Szkoła Główna Handlowa, Oficyna Wydawnicza.

LCA-NN DEEP LEARNING MODEL BASED ON PREDICTION IMPROVEMENT SYSTEM USING MULTIVARIATE DECOMPOSITION

Abstract: In this article, we present a deep learning system that combines MLP neural networks with multivariate decompositions. It is an extension of the system for improving prediction results in a multi-model approach using ICA, PCA, and NMF methods. This can also be considered a method for model aggregation or multidimensional filtering of components that interfere with forecast results. We present the entire process in connection with the blind separation problem, pointing out possible useful inspirations and significant limitations of such interpretations.

Keywords: prediction, deep learning models, multidimensional decompositions, neural networks

JEL classification: C02, C50